A Deeper Understanding: Predicting Review Rating with Text

Ching Lau and Joanna Huang

Abstract

The purpose of this paper is to use user review data to predict sentiment rating and understand how different models are making the prediction.

We first extract narrative features based on our real world understanding from user reviews for the baseline regression models. Next, we build variations of bag-of-words and CNN models. We use integrated gradients to understand the most important words that each model uses for its prediction and then analyze the sensitivity of these models to different words present in the user reviews.

We find that when a model ignores important information in making its prediction, its model performance is more susceptible to changes in input features. Therefore, high accuracy is an indicator of a good model only if the model is picking up the right words for the prediction.

1 Introduction

We implement 6 different models to predict hotel review rating and compare what input features each model focuses on. While we use high accuracy of prediction as our evaluation metric, the aim of this paper is a better understanding of how deep neural networks make predictions and whether it conforms to our real-world understanding.

First, we extract narrative features based on past research in the area using LIWC and NRC Emotion lexicons and NLTK. Then we build two regression models as our baseline. Second, we build two Bag-of-Words models, one for categorical classification and the second for ordinal classification. Third, we build two variations of the CNN model for sentence classification, one with customized word embedding vectors and another with pre-trained GloVec word embeddings.

We hope the insights derived in this paper can be applied to develop a neural network that improves the accuracy of the predictions in this space.

2 Integrated Gradients (IG)

We use an attribution method called Integrated Gradients (IG) (Sundararajan et al., 2017) in [3] to identify words in the user reviews that a deep learning system uses to make a prediction.

Formally, suppose we have a function $F : \mathrm{R}^n \rightarrow [0, 1]$ that represents a deep network, and an input $x = (x_1,...,x_n) \in \mathrm{R}^n$. An attribution of the prediction at input x relative to a baseline input x′ is a vector $A_F(x,x') = (a_1,...,a_n) \in \mathrm{R}^n$ where a_i is the contribution of x_i to the prediction $F(x)$. We can think of F as the probability of a specific response and attributions $a_1,...,a_n$ are the influences/blame-assignments to the input variables $x_1,...,x_n$ on the probability F.

IG is one of the methods to calculate the attributions. We use IG because of its ease and efficiency of implementation and its axiomatic justification. Below is the definition of IG and please refer to (Sundararajan et al., 2017) in [3] for more details.

$$F(x) = F(x') + \textstyle\sum_{i=1} IG_i(x, x')$$

where

$$IG_i(x, x') = (x_i - x'_i) \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} d\alpha$$

Since the attributions are defined relative to a baseline x′, it is best to choose a baseline such that the prediction at the baseline is near zero (F(x′) ≈ 0). In this paper, we use an empty user review as the baseline, that is, a sequence of word embeddings corresponding to the padded value.

We also leverage the visualization method proposed by (Mudrakarka, et al., 2018) in [10] to visualize the attributions of different words in the user reviews to the sentiment rating prediction.

3 Data

For our data, we scrape TripAdvisor reviews from January 2014 to October 2018 for all hotels (excluding B&B and inns and specialty lodging) in San Francisco, New York City, Chicago and

Las Vegas for a total sum of 1,030,236 reviews. We choose this domain and dataset given the massive textual data available as well as the rich metadata such as numeric ratings and geographic location that allow for interesting analysis. In addition to the review text, we use the review's word count, helpfulness rating, user activity level (based on number of contributions) and city as control variables in some of our models.

The dataset is imbalanced with 5%, 6%, 13%, 28%, 49% of reviews from 1-star to 5-star ratings respectively. Therefore, we combine user reviews with 1- & 2-star ratings and create 4 buckets: 2-star, 3-star, 4-star and 5-star ratings for our analysis. For this dataset, we measure the performance of the model by accuracy, precision, recall and F1-score.

The average length of a review is 111 words with more than 98% of reviews less than 500 words. Therefore, for models that required fixed length input, we either pad or truncate the user review to 500 words.

4 Models

4.1 Multinomial Logistic Regression

For our first model, we use a multinomial logistic regression to predict a review's rating. We choose this model as a baseline given its interpretability and the ability to measure coefficient weights to determine feature importance. For each sentence in a review, we extract the features outlined in this section. Then we calculate sentence-level feature percentages and scores to represent each review as a numeric feature vector.

While we recognize that ordinal logistic regression would have been preferable given the natural order in ratings, our data does not satisfy the proportional odds assumption where the relationship between the lowest versus all higher categories of the response variable are the same.

4.1.1 Narrative Elements

Subjectivity and Polarity: We define subjectivity as an expression of some personal feelings, views, or beliefs [4]. Negative experiences are often one-off, personal experiences, thus we hypothesize that negative reviews tend to be more subjective than positive reviews. Polarity is defined as a score to measure the level of positivity and negativity in a review. For these scores, we utilize the existing TextBlob Python library to classify our review sentences. TextBlob utilizes the lexicon en-sentiment.xml which has previously tagged subjectivity and polarity scores for every word. The polarity score is a float within the range [-1.0, 1.0]. The subjectivity is a float within the range [0.0, 1.0] where 0.0 is very objective and 1.0 is very subjective.

Trauma: We seek to test the trauma narrative hypothesis from [1] in the context of hotel reviews. [1] hypothesized that negative restaurant reviews serve as coping mechanisms for dealing with minor traumatic experiences at restaurants. Mirroring their method, we extract linguistic variables to measure negative emotion, narrativity and first person plural. Using LIWC2015, we obtain the percentage of total words within each review that were considered negative emotional words. We count all occurrences of the words "we", "us", "our", "ours", and "ourselves" for first person pronouns. Lastly, for narrativity, we count the past tense and perfect verbs (tagged using NLTK pos tagging), third person pronoun words and additional narrative variables mentioned in [1] such as narrative sequencers and speech act verbs.

Narrative Content: We adapt the methods outlined in [5] for narrative content. According to [5], there are four elements of narrative content: affective and cognitive consciousness and events' spatial and temporal embedding. We only utilize unigrams, rather than trigrams, to determine the level of affect (expression of feeling) and the level of cognition (expression of thought). We view spatial embedding and temporal embedding as tactics of storytelling that users employ to enhance the persuasiveness of their review. We utilize LIWC2015 to obtain the percentage of total words within each review that were categorized under affect, cogproc, spatial and temporal.

Emotional display: As described in [6], service encounters are complex and a customers' satisfaction for the service received is often influenced by a customers' prior emotions. Here we utilize Plutchik's (1980) emotions scale, as it includes Anticipation which has been considered an important factor for a customer's service evaluation. Using the NRC Emotion Lexicon, we count the words in each review that are related to Fear, Anger, Joy, Sadness, Acceptance, Disgust, Anticipation and Surprise.

In total, we extract 20 features for each review. To normalize the features, we convert all counts to percentages. To assess the multicollinearity assumption in our model, we calculate the variance inflation factors for each feature. We remove features with VIF factors above 5, with the exception of city, class, and activity level given their role as control variables with low correlation to the predictors. Spatial and temporal are also kept as variables given they are indicator variables and high VIF's often result from unproportional cases in each category. We also check for adherence to the linearity assumption through graphical means and IRR assumption with likelihood ratio tests that compare logit models with the full set of ratings and with subsets.

Full model results are in Table 3 in the Supplementary Materials. All the coefficients for the features we include are significant at the 1% level. Most features influence the relative probability of being in a 3-, 4- & 5 stars compared to 2-star (note we use 0, 1, 2, 3 in model implementation) in the direction we have foreseen. For example, higher percentage of affect and joy words and higher Polarity score (more positive) is associated with a higher relative probability of a review being a 3-star than a 2-star. Likewise, a higher percentage of disgust and negative emotion is associated with a lower relative probability of a review being a rating 3-star than a 2-star.

Sadness has an influence contrary to intuition. Model results show the higher percentage of sad words, the higher relative probability of being a 3-star than 2-star. Looking at the cases of incorrect predictions, most predicted values differ from their actual value by 1 level. Comparing the feature values to the average of their actual rating, most discrepancies lie in Polarity scores. For example, most reviews that are incorrectly predicted as 2-star and had actual ratings of 3- or 4-star, had below average polarity scores. Slight differences are also seen in past tense and cognition percentages. Incorrect ratings of 5-star tend to have higher than average percentage of affect and joy words.

4.2 Bag of Words (BOW)

4.2.1 BOW Classification

We choose the BOW model because it is simple to implement and has seen great success in text classification. Also, when applying IG to the model to attribute the rating prediction to words in user reviews, the result is intuitive and easy to understand.

The BOW architecture was first proposed in [8]. The model takes user review of length 500 words with the word embedding vector of size 100 and projects the word embedding vectors into the same position (the vectors are summed). It then applies 2 fully connected hidden layers and a softmax output layer to make the prediction.

4.2.1.1. Observations

We find that in general the BOW model is able to pick up most of the important words for the prediction. For example, the top five words that have strong positive attribution for each ratings are:

5-star: excellent, amazing, great, fantastic, love
4-star: good, great, nice, bit, location
3-star: ok, average, not, updating, disappointed
2-star: bad, worst, dirty, terrible, never

However, it is not surprising to see that the model does not account for context (such as, negation, n-gram) of the user reviews. For example, the review below has a 5-star rating but because it includes many negation and negative words (such as, 'old', 'never'), the model predicts a 2-star rating.

ca n't beat the location i ca n't say enough about this place . it feels like a small boutique hotel , but you 'll never know you 're in one of the grand old hotels of michigan 're just a few minutes walk from everything on michigan avenue , and for thead venturous walkers , not too far from millenium park , willis tower , and navy time to explore the old part of the hotel , and check out the pool too .

We also find that some words have a strong negative attribution to a rating that does not match our real world understanding. For example, the word 'good' has a strong negative attribution to a 5-star rating. This inconsistency, we hypothesize, is due to the use of a categorical classification model for a 5-star ordinal rating prediction.

Therefore, we develop a BOW classifier with ordinal categories to see if model will improve accuracy.

4.2.2 BOW Classification with Ordinal Categories

Following the approach proposed by (Chang, et al, 2007) in [9], we modify the BOW classification model architecture to support ordinal categories.

For ordinal categories, if an input vector x belongs to a category k, it is classified automatically to the lower-order categories (1,2,...k-1) as well. The goal is to learn a function to map the input vector x to a probability vector $o = (o_1, o_2, \ldots, o_k, \ldots, o_K)$, where o_i $(i \leq k)$ is close to 1 and o_i $(i \geq k)$ is close to 0. Therefore, we replace the one-hot encodings of the output categories with the following encodings:

```
2-star: [0,0,0]
3-star: [1,0,0]
4-star: [1,1,0]
5-star: [1,1,1]
```

We also replace the output softmax layer to use standard sigmoid function $\frac{1}{1+exp(-z_i)}$ on each output node, without including the outputs from other node, to estimate the probability o_i. The cost function is the squared error of the probability of each output o_i w.r.t. the encoding of its target output t_i.

$$\textit{Cost Function} = \sum_{i=1}^{K}(t_i - o_i)^2$$

Unfortunately, IG does not work as-is when the target output has 3 nodes, so no attribution analysis is done on this model.

4.3 Convolutional Neural Network (CNN)

Since the BOW classifier does not take into account of phrases and terms in the user reviews to make the prediction, we implement a convolutional neural network to see if it is able to predict the rating better.

4.3.1 CNN Architecture

We develop a CNN model similar to the one published by (Kim et al., 2014) in [2]. The model takes user review of length 500 words with the word embedding vector of size 100 and perform convolution operation with filter windows of 2, 3, 4, 5 with 124 feature map each. It then applies a max-over-time pooling operation over the feature map and takes the max value as the feature corresponding to a particular filter. The features are passed to 2 fully connected layer and a softmax output layer to make the prediction.

4.3.2 Model Variations

We implement 2 variations of the model.

CNN-rand: All word embedding vectors are randomly initialized and then modified during training.

CNN-GloVec: A model with pre-trained word vectors from `GloVec`. All words, including the unknown ones that are randomly initialized, are kept static.

4.3.3 Observations

We first analyze whether the CNN model is able to handle negation properly in its prediction by checking the following 2 conditions:

1) the attributions of the negation word and its target word should have the same polarity
2) the attributions of the negation and its target word should support the predicted sentiment rating.

We found that the ability of the CNN model to handle negation is mixed. Here are some examples of negation words are handled properly:

2-star: 'not recommend' ['recommend' has +ve sentiment but the term 'not recommend' is -ve and supports a 2-star rating]

5-star: 'will not stay anywhere' ['not' has -ve sentiment but the term is +ve and supports a 5-star rating]

However, the model fails to handle negation in the following examples where the attributions of the negation word and the target word have different polarities.

5-star: 'never been disappointed'
5-star: 'not disappointed'
3-star: 'not the best hotel'

Second, we check if the model is able to make better predictions with important terms in the user reviews. Below are common terms that have positive attributions for each rating:

5-star: will definitely return, will definitely stay again, highly recommend, amazing experience,
4-star: bit disappointed, good value, good stay, only drawback
3-star: needs upgrading,
2-star: never again, not recommend, worst hotel,

We find that unlike BOW model which has positive or negative attributions on most of the important words, CNN model sometimes ignores some of the important words or terms in making its prediction. For example, the review below has a 5-star rating but the CNN model ignores the important terms (such as 'absolutely one of the best places to stay', 'can't wait to go back') and predicts a 2-star rating.

absolutely one of the best places to stay while in vegas ! absolutely one of the best places to stay while in vegas , friendly staff that treat you like family ! the hotel also has a great security staff to prevent any uninvited or unwanted individuals from wandering in off of the street , cant wait to go back !

Another issue we can see from this example is that, the model does not pick up the contextual information properly. The model focuses on the uninvited individuals instead of the great job that the hotel has done to prevent them from wandering in.

Below is another example the model fails to pick up the contextual information when the review compares the hotel with other hotels. This review has a 5-star rating and the model predicts 3-star.

very modern and clean i have stayed at both venetian and palazzo . the rooms are a bit bigger (and more $) at the venetian .both hotels are entire hotel is very clean , well lit and modern . the housekeeping arrived right away with extra pillows and they cleaned our room as requested while we were out to other hotels we walked through on the strip , we thought these DG(palazzo and venetian) were far and above better hotels . other venues seemed dated , old , dirty , over crowded , lots of smoke : caesars mgm

To understand more contextual information than n-grams can capture, it may need a recursive neural network (RNN) or attention systems which are not explored in this paper.

5 Results

From table 1, we see that the performance of the 4 neural networks are roughly the same even though they are implemented differently.

5.1 Regression vs. BOW

We compare the BOW words that have attributions exceed +/-0.5 for each rating to the features of the multinomial regression to better understand the types of features captured.

Words of negative emotion, sadness, and disgust are positively attributed to reviews with 2- and 3-star. BOW also picks up words of cognition such as 'barely', 'didn't, 'supposed' and 'lacking' which may be related to a discrepancy in expectation and experience. "Gross" seems to be a word reserved for 2-star. The difference between 2-star and 3-star is the noticeable drop in positively associated sad and disgust words. This shift becomes more apparent with the higher ratings as you see more words of Joy and less words of negative emotion and disgust. Cognitive words for 4-star include words like "overall" and "otherwise" which are often used at the end of a review to sum up a satisfactory experienc

Table 1: Model Results

	Accuracy	Precision	Recall	F1-score	Training time per epoch
Multinomial	0.58	0.54	0.58	0.54	-
Prop-odds	0.57	0.51	0.57	0.52	-
BOW	0.72	0.67	0.69	0.68	13:00
BOW-ordinal	0.71	0.67	0.68	0.67	14:38
CNN-rand	0.72	0.68	0.68	0.68	55:25
CNN-GloVec	0.71	0.66	0.66	0.66	41:44

While all reviews have spatial words like “small”, “tiny” and “thin”, higher rating reviews also include words like “large”, “biggest” and “massive”. Likewise, while “noisy” and “dark” are common perception words among all reviews, ‘beautiful’ and ‘quiet’ is only positively attributed to 4-star and 5-star ratings.

Lastly, BOW words for 5-star consist of positive comparatives, such as “coolest” and “best” and time words that may be related to good service such as “fast”, “immediately” and “always”. Looking at the Joy words, BOW is also able to capture words for special occasions such as “graduation”, “celebration”, and “bride.”

Similar to what we find in the multinomial regression, sad words appear to remain prominent even as ratings increase. Upon closer inspection, sad words in 4-star reviews seem to be used for mentioning the “one negative” or “one complaint”’ the customer had. For 5-star reviews, sad words were used in the context of negation (i.e. “cannot fault”, “did not disappoint”).

5.2 BOW vs. CNN

We compare the CNN words that have attributions exceed +/-0.5 for each rating to the regression model and the BOW model and they are similar.

However, from our analysis before, we notice that the CNN model does not always take into account of all the important terms in making the prediction. Therefore, we check how sensitive the CNN model is to the exposure of different words in the user reviews.

We follow the methods proposed by (Mudrakarta, et al., 2018) in [10] to perform a sensitivity analysis on the BOW and CNN models. We prepend words from the most common words for 2- & 5-star ratings from both models to the test data and check how much it affects the model accuracy.

From Table 2, we see that the accuracy of the BOW and CNN models perform fairly consistent when the prefix is neutral. However, when the prefix has positive or negative sentiment, the accuracy of the CNN model drops more than the BOW model.

To improve the CNN model, increasing the model complexity like adding more filters and training with more data and augmenting the dataset with more adversarial examples will help

Table 2: Sensitivity Analysis

Prefix	BOW accuracy	CNN accuracy
in not many words	0.71 (↓ 0.3%)	0.71 (↓ 0.2%)
amazing experience, perfect, love it	0.65 (↓ 9.15%)	0.56 (↓ 21.6%)
dirty, rude, worst experience	0.52 (↓ 26.6%)	0.54 (↓ 24.7%)

6 Discussion

Here are other patterns we observe:

- Review rating is not an objective score. We find that users’ choice of words to describe their experience of the same review rating are different. Future work can take user information into account as proposed by (Tang et al., 2015) in [11]. We also found many examples where the user reviews do not match their ratings.
- Hyperparameter tuning such as, embedding vector size and architecture complexity may have a big impact on the model performance and should be explored.
- The TripAdvisor dataset is biased towards 4- & 5-star ratings. We can explore different bucketing methods or under-sample reviews with 4- & 5-star ratings.
- We find that user reviews have a lot of spelling mistakes. The size of the vocab decreases from ~57K to ~38K after simple spell check. However, the models in this paper are not performed on spell-

checked dataset which can be explored in the future.

- Closer validation of the lexicons is recommended in the future to consolidate category overlaps (i.e. LIWC and NRC both have 'Anger' words) and correct word inconsistencies (i.e. lovely categorized as 'sad' by NRC).

7 Conclusion

We implement 6 models to predict the sentiment ratings from user reviews. In our analysis, the BOW model seems superior to other models. It picks up most of the important words in the user reviews that matches our real world understandings for its prediction. It is also slightly more consistent to changes to input features than the more complex CNN model. In terms of performance, the BOW model is comparable to the CNN model and it takes less than 25% of the time to train. Though the CNN model that we implement is not state-of-the-art and has a lot of rooms for improvement, we believe the BOW model is a good simple model that performs competitively. Therefore, we recommend to have the BOW model as a baseline model for future classification models.

References

[1] Jurafsky, Dan et al. Narrative framing of consumer sentiment in online restaurant reviews. First Monday, [S.l.], mar. 2014. ISSN 13960466. https://firstmonday.org/ojs/index.php/fm/article/view/4944/3863>. Date accessed: 25 nov. 2018. doi:https://doi.org/10.5210/fm.v19i4.4944.

[2] Kim, Yoon. Convolutional Neural Networks for Sentence Classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1746–1751, October. https://www.aclweb.org/anthology/D14-1181

[3] Sundararajan, Mukund et al. Axiomatic Attribution of Deep Networks. 13 June 2017. https://arxiv.org/pdf/1703.01365.pdf

[4] Liu, Bing. 2010. Subjectivity and Emotion. In *Sentiment analysis and Opinion Mining* p. 19. http://stp.lingfil.uu.se/~santinim/sais/2014/bingliu.pdf

[5] Van Laer, Tom et al. What Happens in Vegas Stays on TripAdvisor? A Theory and Technique to Understand Narrativity in Consumer Reviews.

[6] Gountas, Sandra et al. The Influence of Consumers' Emotions on their Service Product Evaluation. In *CAB International 2004. Consumer Psychology of Tourism, Hospitality and Leisure, Volume 3 (G.I. Crouch, R.R. Perdue, H.J.P. Timmermans and M. Uysal).* Chapter 3.

[7] Mohammad, Saif. NRC Word-Emotion Association Lexicon. https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm

[8] Mikolov, Tomas. 2013. Efficient Estimation of Word Representations in Vector Space. https://arxiv.org/abs/1301.3781

[9] Cheng, Jianlin. 2007. A Neural Network Approach to Ordinal Regression. https://arxiv.org/abs/0704.1028

[10] Mudrakarta, Pramod. 2018. Did the Model Understand the Question? http://aclweb.org/anthology/P18-1176

[11] Tang, Duyu. 2015. User Modeling with Neural Network for Review Rating Prediction. https://www.ijcai.org/Proceedings/15/Papers/193.pdf

8 Supplementary Materials

Table 3. Multinomial Logistic Regression

MNLogit Regression Results

Dep. Variable:	Rating	No. Observations:	721165
Model:	MNLogit	Df Residuals:	721111
Method:	MLE	Df Model:	51
Date:	Fri, 07 Dec 2018	Pseudo R-squ.:	0.2391
Time:	20:18:12	Log-Likelihood:	-6.6141e+05
converged:	False	LL-Null:	-8.6929e+05
		LLR p-value:	0.000

Rating=1	coef	std err	z	P>|z|	[0.025	0.975]
Word_Count	-0.0010	3.79e-05	-25.395	0.000	-0.001	-0.001
Helpful	0.0007	0.000	4.687	0.000	0.000	0.001
city	-0.0238	0.005	-4.925	0.000	-0.033	-0.014
class	-0.0037	0.001	-6.074	0.000	-0.005	-0.002
activity_level	0.4671	0.012	38.748	0.000	0.443	0.491
Third Person	-6.1622	0.327	-18.862	0.000	-6.803	-5.522
First Person	-0.9601	0.287	-3.348	0.001	-1.522	-0.398
negemo	-1413.3091	30.299	-46.645	0.000	-1472.694	-1353.924
affect	1018.4648	22.946	44.385	0.000	973.491	1063.439
Joy	11.1966	0.380	29.456	0.000	10.452	11.942
Disgust	-21.4653	0.568	-37.760	0.000	-22.579	-20.351
Sad	5.6435	0.494	11.413	0.000	4.674	6.613
Narrative Seq	-16.7271	0.513	-32.594	0.000	-17.733	-15.721
Past Tense	-1.3099	0.143	-9.142	0.000	-1.591	-1.029
spatial	-0.0923	0.010	-9.076	0.000	-0.112	-0.072
temporal	-0.0975	0.010	-9.411	0.000	-0.118	-0.077
Polarity2	4.8020	0.052	92.571	0.000	4.700	4.904
cogproc	-83.3272	10.467	-7.961	0.000	-103.842	-62.812

Rating=2	coef	std err	z	P>|z|	[0.025	0.975]
Word_Count	-0.0012	3.88e-05	-30.338	0.000	-0.001	-0.001
Helpful	0.0007	0.000	5.040	0.000	0.000	0.001
city	-0.0886	0.005	-18.256	0.000	-0.098	-0.079
class	0.0103	0.001	16.834	0.000	0.009	0.011
activity_level	0.5405	0.012	44.979	0.000	0.517	0.564
Third Person	-10.7316	0.337	-31.860	0.000	-11.392	-10.071
First Person	6.5081	0.284	22.901	0.000	5.951	7.065
negemo	-2635.8432	39.573	-66.608	0.000	-2713.404	-2558.282
affect	1358.0446	22.978	59.101	0.000	1313.008	1403.081
Joy	16.4106	0.376	43.598	0.000	15.673	17.148
Disgust	-52.6651	0.683	-77.141	0.000	-54.003	-51.327
Sad	12.4239	0.517	24.016	0.000	11.410	13.438
Narrative Seq	-28.5701	0.534	-53.529	0.000	-29.616	-27.524
Past Tense	-6.1296	0.148	-41.504	0.000	-6.419	-5.840
spatial	-0.1501	0.010	-14.666	0.000	-0.170	-0.130
temporal	-0.0256	0.010	-2.444	0.015	-0.046	-0.005
Polarity2	10.0145	0.055	182.540	0.000	9.907	10.122
cogproc	-541.2222	11.280	-47.980	0.000	-563.331	-519.114

Rating=3	coef	std err	z	P>|z|	[0.025	0.975]
Word_Count	-0.0020	4.18e-05	-47.639	0.000	-0.002	-0.002
Helpful	0.0009	0.000	6.025	0.000	0.001	0.001
city	-0.2155	0.005	-43.948	0.000	-0.225	-0.206
class	0.0461	0.001	74.834	0.000	0.045	0.047
activity_level	0.0123	0.012	0.998	0.318	-0.012	0.036
Third Person	-12.8464	0.340	-37.822	0.000	-13.512	-12.181
First Person	15.0463	0.283	53.153	0.000	14.492	15.601
negemo	-3942.6397	45.057	-87.503	0.000	-4030.950	-3854.329
affect	1372.2477	22.997	59.671	0.000	1327.175	1417.321
Joy	17.3296	0.376	46.044	0.000	16.592	18.067
Disgust	-63.7707	0.719	-88.692	0.000	-65.180	-62.361
Sad	9.6153	0.526	18.283	0.000	8.585	10.646
Narrative Seq	-26.1042	0.533	-48.990	0.000	-27.149	-25.060
Past Tense	-9.8384	0.149	-65.916	0.000	-10.131	-9.546
spatial	-0.2107	0.010	-20.500	0.000	-0.231	-0.191
temporal	0.0164	0.011	1.551	0.121	-0.004	0.037
Polarity2	12.5791	0.055	226.719	0.000	12.470	12.688
cogproc	-810.8656	11.497	-70.530	0.000	-833.399	-788.332